//消息类型

typedef enum \_MsgType{

//...

}MsgType;

//文件后缀格式

typedef enum \_FileSuffixType{

NONE\_TYPE = 0, //没有格式

TXT\_TYPE, //文本格式

PDF\_TYPE, //pdf格式

JPG\_TYPE //图片格式

//....

}FileSuffixType;

//文件信息

typedef struct \_FileInfo{

FileSuffixType m\_eFileSuffixType; //文件后缀

unsigned long m\_ulFileLen; //文件长度

char m\_aFileName[4096]; //文件名字

}FileInfo;

//文件分页

typedef struct \_FilePagingInfo{

unsigned long m\_ulCurrentPosition; //当前位置

unsigned long m\_ulRequestLen; //请求长度

FileInfo m\_stFileInfo; //文件信息

char m\_aRequestFileData[0]; //请求数据

}FilePagingInfo;

typedef struct \_FileMsg{

MsgType m\_eMsgType; //消息类型

int m\_iMsgLen; //消息长度

char m\_aMsgData[0]; //消息内容

}FileMsg, \*pFileMsg;

首先定义上面几个结构体，客户端发送请求报文FilePagingInfo，服务端根据ulCurrentPosition和ulRequestLen调动文件指针，然后返回m\_aRequestFileData。 文本和图片支持这种方法，pdf，doc，excel等其他格式未实验。

